Bi-directional Linkability From Wikipedia to Documents and Back Again: UMass at TREC 2012 Knowledge Base Acceleration Track

Jeffrey Dalton
University of Massachusetts, Amherst
jdalton@cs.umass.edu

Laura Dietz
University of Massachusetts, Amherst
dietz@cs.umass.edu

1 Introduction

This notebook details the participation of the University of Massachusetts Amherst in the Cumulative Citation Recommendation task (CCR) of the TREC 2012 Knowledge Base Acceleration Track. UMass' objective is to introduce a single model for Knowledge Base Entity Linking and Knowledge Base Acceleration stream filtering using bi-directional linkability between knowledge base (KB) entries and mentions of the entities in documents.

Our system focuses on estimating linkability between documents and Knowledge Base entities which measures compatibility in two directions: (1) from a KB entity to documents and (2) from mentions of entities in documents to their KB entries. The entity to document direction, is modeled as a retrieval task where the goal is to identify the most relevant documents for an entity in the evaluation time range. However, if the entity is ambiguous, the retrieved documents may contain documents that are relevant to other entities with the same or similar name. To address this, we want to leverage information from the document to disambiguate the entity. We observe that this problem, from mention to KB entity, is very similar to the TAC Knowledge Base Population Entity Linking Task (Ji et al., 2011). The major goal of our participation is to explore how these two directions, from KB to documents and back can be combined.

For KBA, the goal is to identify documents from a stream that are central for a given entity in Wikipedia. Some participants viewed this as a classification problem and trained supervised classifiers for each entity. Instead, our approach to the problem is based upon document ranking. We combine probabilistic information retrieval and then combine the results with TAC entity linking for re-ranking and filtering.

Our ranking approach has three stages: First, documents are retrieved from the stream corpus using an entity query model. Second, potential mentions of the target entity are identified in the retrieved documents. Third, links between the document mentions and the target entity are established or dismissed, giving rise to a filtered (or re-ranked) list of results that mention the target entity. Our initial system leverages the bi-directional relevance as a simple form of re-ranking of retrieved documents, but we envision tighter integration in the future.

The baseline retrieval run generates a query from the Wikipedia KB entry, including the name, name variants, and linked entities. We also incorporate contextual evidence from the document stream by using the documents in the training time period as relevance feedback documents. We use Latent Concept Expansion (Metzler and Croft, 2007) to estimate important contextual words and NER concepts.

Our experiments show that incorporating entity context from query expansion methods provides significant gains both in precision and recall over the baseline, with all of our experimental runs outperforming the median. Our best performing run incorporates linkability evidence from the TAC Entity Linking model.

2 Method

Our method to estimate bi-directional linkability uses graphical latent variable models that combine probabilistic retrieval and extraction models. In each direction, we first generate a high recall set of candidates using the Markov Random Field retrieval model (Metzler and Croft, 2005) to construct a query model that includes a model of entity context. The retrieval model includes name variations, surrounding words and NER spans which are identified from text associated with the target entity. We experiment with various methods for estimating the model of an entity, using Latent Concept Expansion (LCE) to incorporate across-document evidence from the corpus. The result is a focused set of candidate documents and knowledge base entries, ranked by the likelihood of referring to the same entity. This set is either used directly, or acts as input to more advanced

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **NOV 2012**	2. REPORT TYPE	3. DATES COVERED **00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE **Bi-directional Linkability FromWikipedia to Documents and Back Again: UMass at TREC 2012 Knowledge Base Acceleration Track**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Massachusetts, Amherst,Department of Computer Science,Amherst,MA,01003**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the Twenty-First Text REtrieval Conference (TREC 2012) held in Gaithersburg, Maryland, November 6-9, 2012. The conference was co-sponsored by the National Institute of Standards and Technology (NIST) the Defense Advanced Research Projects Agency (DARPA) and the Advanced Research and Development Activity (ARDA). U.S. Government or Federal Rights License

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**6**	

inference methods.

3 Corpus Processing

Our retrieval models are implemented using Galago[1], an open source retrieval engine which is part of the Lemur toolkit. Galago supports indexing of large scale data in a distributed cluster environment with a MapReduce-like framework called TupleFlow.

Both the KBA stream corpus and the Wikipedia knowledge base are indexed to efficiently support bi-directional linking queries.

3.1 KBA Stream Corpus

We used the "cleansed" documents with NER information from the KBA stream corpus. These documents are indexed with Galago, stripping out HTML tags. No stemming or stopword removal is performed. In order to filter the stream by time stamp and source type (e.g. linking, social, news), we index this information in Galago fields. Further NER information is preserved in the documents, to be used in relevance feedback queries.

For efficiency we create a separate index shard for each month. Indexing each shard took between four and eight hours on a cluster of fifty nodes. Per-shard collection statistics are given in Table 1.

3.2 Wikipedia Knowledge Base

For entity linking, we use a Freebase Wikipedia Extraction (WEX) dump of English Wikipedia from January 2012 which provides the Wikipedia page in machine-readable XML format and relational data in tabular format. The Freebase dump contains 5,841,791 entries. We filter out non-article entries, such as category pages. The resulting index contains 3,811,076 documents and over 60 billion words.

The goal is to create an index with fields for: anchor text (internal as well as from the Web), Wikipedia categories, Freebase names, Freebase types, redirects, article titles, and full-text for each article. Most of this information is contained in the WEX dump. We also incorporate external web anchor text to Wikipedia entries using the the Google Cross-Wiki dictionary (Spitkovsky and Chang, 2012), which contains 3 billion links and 297 million associations from 175 million unique anchor text strings.

4 KB Entities to Documents

For each target entity from Wikipedia, the first step is to retrieve a high recall set of stream documents. First, name variants and potentially disambiguating context is extracted from the target's Wikipedia article. We leverage the stream corpus to re-weight disambiguating context. From these ingredients, we build a retrieval query against the stream corpus.

The goal is to identify:

- the target entity's name,
- name variants by which the entity is referred to,
- contextual words,
- related named entities.

4.1 Extracting Name Variants and Disambiguating Context

The canonical name of the target entity is taken from the title of the Wikipedia article. Name variants for the Wikipedia entry are gathered from the title, redirects, Freebase names, disambiguation links, and incoming anchor text. Related named entities are taken from titles of in- and outlinks of the target's Wikipedia page.

4.2 Entity Modeling using Latent Concept Expansion

We estimate disambiguating context from the corpus and training document evidence using Latent Concept Expansion (LCE). LCE is a query expansion technique for estimating contextual evidence built upon the Markov Random Field retrieval framework. We use it to model dependencies between related entities by including NER name spans as types of concepts. LCE estimates the salience of an entity span from documents that are relevant to the target entity. The intuition is that the salience of words and named entities increases the more often they occur in documents relevant to the target entity.

For relevance feedback, we use the set of relevant documents from the pre-cutoff sample documents. In one experimental run we also add post-cutoff documents using pseudo-relevance feedback. We now discuss the individual components of the expansion model. The most probable k unigrams (after removing stopwords) are used as disambiguating contextual words with weights ϕ^{CW}. We now discuss how we estimate the NER ϕ^{NER} weights. The first source of spans are the named entities mentioned in the inlink and outlink structure. In addition, we use NER spans that frequently occur in the relevant documents. After sets of entity spans from the KB entry and external documents are combined, the top k weighted named entities are selected as context. We use both NER spans from the Wikipedia entry and external spans from relevant documents because they capture different aspects of relevance for the entity. The corpus may be biased towards one event in time. The link information from Wikipedia captures long-term hand-constructed relationships, but they may not be timely. The current method for determining entity context importance is a first step. In the future, we plan to experiment with more advanced

[1] http://www.lemurproject.org/galago.php

Month	Documents	Collection Length	Index Size (GB)	Total Size (GB)
October 2011	36,547,282	54,33,597,431	22	245
November 2011	55,434,234	14,529,421,474	55	673
December 2011	62,773,692	16,058,713,120	62	739
January 2012	60,799,418	16,983,265,272	64	781
February 2012	58,147,836	18,488,791,637	67	833
March 2012	50,857,928	19,388,982,395	67	871
April 2012	33,796,674	14,217,201,526	51	835
May 2012	395,732	447,158,725	1	21
Total	358,752,796	100,113,534,149	389	4998

Table 1: KBA Galago Shard Statistics

```
#combine:0=$\lambda^{\text{T}}$:1=$\lambda^{\text{NV}}$:2=$\lambda^{\text{CW}}$:3=$\lambda^{NER}$(
  #seqdep($T$)
  #combine(#seqdep($nv_0$)...#seqdep($nv_k$))
  #combine:0=$\phi_0^{\text{CW}}$ :... $k$=$\phi_k^{\text{CW}}(cw_0, \ldots, cw_k)$)
  #combine:0 $= \phi_0^{\text{NER}}$ : ... $k = \phi_k^{\text{NER}}$(
    #seqdep($ner_0$), ..., #seqdep($ner_k$)
  )
)
```

Figure 1: LCE query for retrieving relevant stream documents in Galago query syntax. The query includes the entity name (T), name variants (NV), context words (CW), and NER spans (NER).

techniques for combining and weighting the evidence from the local Wikipedia document with the external evidence from the document collection.

4.3 Retrieving Relevant Stream Documents

The entity model, M_E we use for retrieving stream documents consists of four concept types in K: the entity name T, name variations NV, context words CW, and context NER spans NER. For each entity we compute the score for an entity, E and a document D as follows.

The query model scores the documents in the collection using a log-linear weighted combination of the matches of the concepts K and ranks the documents using this score.

$$sc(E,D) = \sum_{t \in T} \lambda_t \sum_{k \in K} \phi_k \psi(k,D) \tag{1}$$

The potential function $\psi(k,D)$ is a real valued score for a concept in a document, which itself may be a submodel. In M_E we use a sequential dependence model to estimate $\psi(k,D)$ for all concept types, except CW which consists only of unigrams. For each scoring component we use the matching function in Equation 2. We set the $\lambda^{\text{T}}$and λ^{NV} parameters for both the name and name variants to be constant. For the context word and NER span weights, we estimate the latent ϕ_k parameter weights using relevance model weighting (Lavrenko and Croft, 2001).

The score of a concept in document is the log of the probability of a concept, k, given a document D with Dirichlet smoothing, i.e.,

$$f(k,D) = \log \frac{tf_{k,D} + \mu \frac{tf_{k,C}}{|C|}}{|D| + \mu} \tag{2}$$

where $tf_{k,D}$ is the number of occurrences of the concept in the document, $tf_{k,C}$ is the number of occurrences in the collection, $|D|$ is the number of terms in document, $|C|$ is the number of words in the collection, and μ is the smoothing parameter that is set empirically.

The query model is run using the Galago search engine to score all of the stream documents. The query is given in Galago's query syntax in Figure 1. The result of retrieval is a linkability score in the direction of Wikipedia entity to documents which can be used as-is or re-ranked further. The source code for our KBA system including Galago configuration will be available online [2].

5 Entity Mentions to Wikipedia

We estimate linkability in the opposite direction from document mention to Wikipedia entity using a linking model developed for the TAC KBP Entity Linking task. For details of the TAC system, including features please see our TAC 2012 notebook paper (Dietz and Dalton, 2012).

5.1 Identifying Mentions of the Target Entity

For each candidate document retrieved for a target entity, we extract potential mentions of that entity. Across the set of all mentions, we identify the mention with the highest

[2] `http://www.github.com/CIIR/TrecKBA`

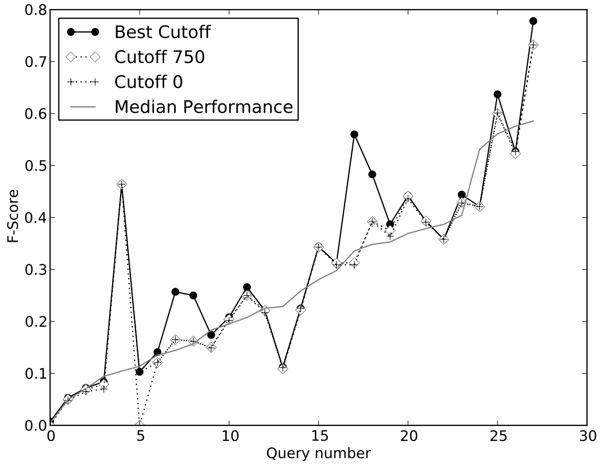

Figure 2: F-Score performance over queries at different cutoff thresholds. Queries are sorted by difficulty in terms of median F-score.

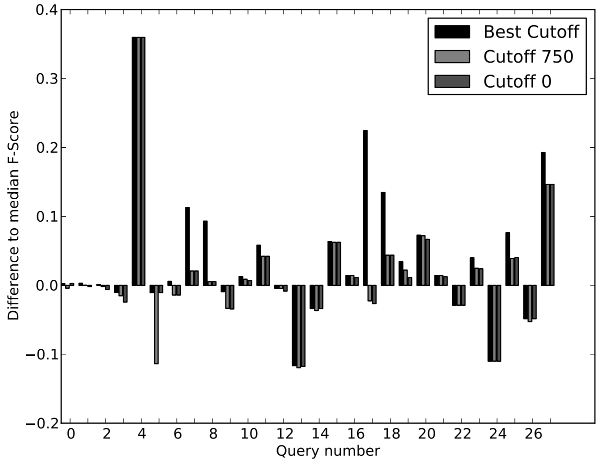

Figure 3: Difference in F-score to the median performance over queries.

similarity to target entity by searching for the target entity name and name variants. Matches are identified with string matches ignoring case and punctuation, preferring exact matches and high confidence name variants over partial name matches. If no matching entity mentions are found, a dummy empty mention is created.

5.2 Re-ranking Mentions to Match the Target Entity

Next, each of the canonical entity mentions is linked against Wikipedia entries — which is the direction evaluated in the entity linking task of the TAC KBP competition. We train a supervised discriminative ranking model with TAC entity linking data from years 2010 and 2011. It incorporates features based on string similarity of names, similarity of term vectors, and name confidence based on ambiguity of anchor texts. A full list of features and a complete description of the entity linking system is provided in our TAC KBP notebook paper.

6 Experimental Results

6.1 Setup

We now describe the parameter setting used for the model. For scoring with Equation 2 we use the default smoothing value, $\mu = 2000$. It is important to note that we only used background term statistics from the training time range. For the free parameters in our Sequential Dependence (SD) sub-models we estimate the parameters using training data from the TAC KBP 2010 entity linking data, resulting in settings $\lambda_{T_D} = 0.29$, $\lambda_{O_D} = .21$, and $\lambda_{U_D} = 0.50$. These parameters place greater emphasis on the ordered window and term proximity, which is logical since the queries consist largely of names. We use the LCE context model to retrieve and rank the stream documents. We manually set the concept type weights as: $\lambda_T = 0.3$, $\lambda_{NV} = 0.3$, $\lambda_{CW} = 0.2$, $\lambda_{NER} = 0.2$. These parameter setting are similar to the default LCE settings, which provides half the weight to the original query and half to expansion terms.

The result of running the query is an unnormalized log probability. To produce a score in the 1 to 1000 confidence range, we normalize the score by the sum of the retrieved document scores. We scale this approximation of the posterior probability by 1000. We experimented with other normalization techniques because we hypothesized that this would affect the optimal cutoff, but as we see in Figure 3 the cutoff appears to have little impact on the evaluation results.

6.2 Run comparison

In this section, we compare the runs submitted to the CCR task of the TREC 2012 KBA Track. The runs we submitted are variations of the models described previously. The descriptions of the runs follow:

1. NV Full Stream – This a baseline run using the entity name and name variations only, scoring the full stream (TTR + ETR) documents , with $\lambda_T = 0.5$, $\lambda_{NV} = 0.5$ $\lambda_{CW} = 0$, $\lambda_{NER} = 0$. The highest scoring 6000 documents are returned by the run. (submitted run ID:FS_NV_6000)

2. NV – This run uses entity name and name variations only, scoring the post-cutoff (ETR) documents , with $\lambda_T = 0.5$, $\lambda_{NV} = 0.5$ $\lambda_{CW} = 0$, $\lambda_{NER} = 0$. The highest scoring 1500 documents are returned by the run. (submitted run ID: PC_NV_150050)

3. LCE10 – This run employs explicit relevance feed-

back on the TTR documents using Latent Concept Expansion to estimate related context words (CW) and NER names (NER) using 10 context words and 10 NER spans each from Wikipedia and the training documents. The parameter setting used are: $\lambda_T = 0.3$, $\lambda_{NV} = 0.3$, $\lambda_{CW} = 0.2$, $\lambda_{NER} = 0.2$. The highest scoring 1500 documents are returned by the run. (submitted run ID:PC_RM10_150050)

4. LCE20 – This run employs explicit relevance feedback on the TTR documents using Latent Concept Expansion to estimate related context words (CW) and NER names (NER) using 20 context words and 20 NER spans each from Wikipedia and the training documents. The parameter setting used are: $\lambda_T = 0.3$, $\lambda_{NV} = 0.3$, $\lambda_{CW} = 0.2$, $\lambda_{NER} = 0.2$. The highest scoring 1500 documents are returned by the run. (submitted run ID: PC_RM20_150050)

5. LCE10+TAC – This run uses LCE10 to retrieve a candidate set of results. Then, TAC entity linking queries are generated from the entities mentioned in the candidates. The supervised TAC linker is applied and the results re-ranked with respect to the target entity. The highest scoring 1500 documents are returned by the run. (submitted run ID:PC_RM10_TACRL50)

6. LCE10 + TAC + PRF – The goal of this run is to improve recall using pseudo-relevance feedback (PRF) over the entire post cutoff stream. The initial query is generated from relevance feedback using LCE on the pre-cutoff training documents using the results from LCE10. Then, the top 50 retrieved documents are re-ranked using the TAC entity linking supervised ranker. The highest scoring 10 documents are used to generate a PRF query model over the post-cutoff (ETR) document set. The PRF query model uses The parameter settings: $\lambda_T = 0.3$, $\lambda_{NV} = 0.3$, $\lambda_{CW} = 0.2$, $\lambda_{NER} = 0.2$. The highest scoring 2000 documents are returned. (completed after deadline)

A summary of the results are shown in Table 2. The LCE context models outperform using name variants only. Additional small improvement is made applying the TAC supervised ranking model to results retrieved using LCE. It does not appear that pseudo-relevance feedback using the evaluation time documents provided any additional benefit. This seems to indicate context models using only the training documents are just as effective as models incorporating evidence from the full stream. Overall, our best performing model LCE10+TAC combines bi-directional evidence from LCE with re-ranking using the TAC entity linking model.

Method	Best F-Score
NV Full Stream	0.277
NV	0.274
LCE10	0.298
LCE20	0.293
LCE10+TAC	**0.305**
LCE10+TAC+PRF	0.299
TREC Avg. Median	0.289

Table 2: Comparison of Best F-Score of the runs. Best result appears in boldface.

6.3 Further Analysis

We examine the query-by-query performance of the our top performing run, LCE10+TAC model in Figure 2 and Figure 3 and how it compares with other teams. The results show that for our optimal cutoff over 68.9% of our queries are above the median. However, if our overall best average cutoff is used, 55.2% of queries are above the median. Our best performing queries are Basic_Element_(music_group), Jim_Steyer, Nassim_Nicholas_Taleb, and James_McCartney. The worst performing queries in order are Basic_Element_(company), Boris_Berezovsky_(businessman), Satoshi_Ishii, Darren_Rowse, and William_D._Cohan. All the cutoff values correlate highly, with 750 and cutoff 0 both perform comparably despite retrieving very different numbers of results. Choosing a particular cutoff value to evaluate is difficult. The reasons for the similar effectiveness across cutoffs is unclear, but we hypothesize that it may be caused by the evaluation process where only judged negative documents are counted as false positive examples.

In retrospect, the performance of our runs would have improved if more documents (>1500) were retrieved. The NV Full Stream run that retrieves six thousand documents over both the ETR and TTR periods outperforms same method that retrieves only fifteen hundred documents on the evaluation time range. Instead of returning thousands of potentially relevant documents, we focus on ranking a smaller set of highly relevant results. In the future, rank based evaluation metrics may be used to further characterize the behavior.

7 Conclusions

In our submissions to KBA we experimented with bi-directional linkability between Wikipedia and documents to estimate centrality. We attempted to combine evidence from both directions: from an entity to documents and back. Our goal was to utilize the TAC entity linking to reduce noise and improve precision for ambiguous entities.

We present a single model that uses graphical latent variable models with probabilistic retrieval to generate a focused set of candidates, rank the results, and combine evidence from cross-document entity context using relevance feedback. Our experiments show that incorporating evidence from mention to entity using a TAC linker is a promising area and may improve over models that only use evidence in one direction from entities to documents.

One area for future work is modeling temporal change for the entity. We do not explicitly model the temporal change in the stream structure of the KBA corpus. However, we note that there is significant recent work using temporal relevance feedback (Keikha et al., 2011) that could be built upon for this task.

The current use of bi-directional information in our model is limited. The TAC entity linking model is used mainly as a re-ranker. Combining these two tasks is challenging because the linkability evidence is not symmetric, with different sources of evidence in either direction. We intend to explore ways of modeling context and combining these two linkability directions further in our future submissions.

8 Acknowledgment

This work was supported in part by the Center for Intelligent Information Retrieval, in part under subcontract #19-000208 from SRI International, prime contractor to DARPA contract #HR0011-12-C-0016, and in part by NSF grant #IIS-0910884. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsor.

References

Laura Dietz and Jeffrey Dalton. 2012. Across-Document neighborhood expansion: UMass at TAC KBP 2012 entity linking. In *Proceedings of the Text Analysis Conference (TAC KBP)*.

Heng Ji, Ralph Grishman, and Hoa Dang. 2011. Overview of the TAC2011 knowledge base population track. In *Text Analysis Conference*.

M. Keikha, S. Gerani, and F. Crestani. 2011. Temper: A temporal relevance feedback method. *Advances in Information Retrieval*, pages 436–447.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 120–127, New York, NY, USA. ACM.

Donald Metzler and W. Bruce Croft. 2005. A markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 472–479, New York, NY, USA. ACM.

D. Metzler and W.B. Croft. 2007. Latent concept expansion using markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 311–318. ACM.

Valentin I. Spitkovsky and Angel X. Chang. 2012. A Cross-Lingual dictionary for english wikipedia concepts. In *Conference on Language Resources and Evaluation*.